What is claimed is:

- 1 1. A method for reducing test data volume in the testing of logic products,
- 2 comprising the steps of:
- 3 (a) generating original test vector data including care bits and non-care bits;
- 4 (b) filling said non-care bits with a repeated value to form a highly
- 5 compressible test vector data set; and
- 6 (c) compressing said highly compressible test vector data set to form a
- 7 compressed test vector data set.
- 1 2. The method of claim 1, further comprising the steps of:
- 2 transmitting said compressed test vector data set to a test system; and
- 3 recovering the care bits of said original test vector data from said
- 4 compressed vector data set, for loading into input latches of a tester in said test
- 5 system.
- 1 3. The method of claim 1, wherein said step (b) comprises:
- 2 generating a background vector data set; and
- forming a differential vector data set by XORing said care bits with
- 4 corresponding bits in said background vector data set.
- 1 4. The method of claim 3, wherein said XORing sets a substantial portion of
- 2 said care bits to a value of 0 in said differential vector data set.
- 1 5. The method of claim 3, further comprising the step of attaching a header to

- 2 said differential vector data set, said header identifying an algorithm and seed used
- 3 to generate said background vector data set, wherein said differential vector data
- 4 set with attached header form said highly compressible test vector data set.
- 1 6. The method of claim 5, wherein said recovering step comprises:
- 2 decompressing said compressed test vector data set;
- 3 extracting said differential vector data set and attached header;
- 4 reconstructing said background vector data set from said header; and
- 5 XORing said reconstructed background vector data set with said extracted
- 6 differential vector data set to form a reconstructed test vector data set.
- 1 7. The method of claim 6, wherein said reconstructed test vector data set
- 2 comprises the care bits of the original test vector data, with the non-care bits
- 3 having the values of the corresponding background vector data bits.
- 1 8. The method of claim 3, wherein said background vector data set comprises
- 2 a random distribution of bits having values of both "0" and "1".
- 1 9. A method for reducing test data volume in the testing of logic products,
- 2 comprising the steps of:
- 3 generating redundant test vectors from original test vector data; and
- 4 utilizing a repeat capability of a tester to load input latches of the tester with
- 5 the redundant vectors.
- 1 10. The method of claim 9, wherein said step of generating redundant test

- 2 vectors comprises repeating a care bit value encountered in a first test vector of
- 3 said original test vector data in non-care bits of neighboring test vectors.
- 1 11. The method of claim 10, wherein said original test vector data comprises a
- 2 matrix of test vectors arranged in rows and columns, and said care bit value is
- 3 repeated in the same column for each row of said matrix, until a different care bit
- 4 value is encountered.
- 1 12. The method of claim 11, wherein when said different care bit value is
- 2 encountered, said different care bit value is repeated in the same column for each
- 3 row of said matrix, until another different care bit value is encountered.
- 1 13. A computer-usable medium storing computer-executable instructions, said
- 2 instructions when executed implementing a process for reducing test data volume
- 3 in the testing of logic products, comprising the steps of:
- 4 (a) in original test vector data comprising care bits and non-care bits, filling
- 5 said non-care bits with a repeated value to form a highly compressible test vector
- 6 data set; and
- 7 (b) compressing said highly compressible test vector data set to form a
- 8 compressed test vector data set.
- 1 14. The computer-usable medium of claim 13, said process further comprising:
- 2 transmitting said compressed test vector data set to a test system; and
- 3 recovering the care bits of said original test vector data from said
- 4 compressed vector data set, for loading into input latches of a tester.

- 1 15. A program storage device readable by a machine, tangibly embodying a
- 2 program of instructions executable by the machine to perform method steps for
- 3 generating redundant test vectors from original test vector data, by repeating a care
- 4 bit value encountered in a first test vector of said original test vector data in non-
- 5 care bits of neighboring test vectors.
- 1 16. The program storage device of claim 15, wherein said original test vector
- 2 data comprises a matrix of test vectors arranged in rows and columns, and said
- 3 care bit value is repeated in the same column for each row of said matrix, until a
- 4 different care bit value is encountered.
- 1 17. A method comprising:
- 2 (i) forming a compressed test data set by setting care bits and non-care bits
- 3 in original test data to a repeated value and compressing said original test data;
- 4 (ii) downloading said compressed test data set to a testing system for logic
- 5 products;
- 6 (iii) decompressing said compressed test data set to form a decompressed
- 7 test data set; and
- 8 (iv) loading input latches in a logic product with said decompressed data
- 9 set.
- 1 18. The method of claim 18, wherein said step (i) comprises XORing said care
- 2 bits with corresponding bits of a background vector data set.

- 1 19. The method of claim 18, wherein said background vector data set has a
- 2 random distribution of bits having values of 0 and 1.
- 1 20. The method of claim 19, wherein said XORing sets a substantial portion of
- 2 said care bits to a value of 0.
- 1 21. A method for testing logic products comprising:
- 2 applying a first testing technique to said logic products during a first testing
- 3 period, said first testing technique comprising loading input latches of a tester for
- 4 testing said logic products with test vectors comprising a random distribution of
- 5 bits; and
- applying a second testing technique to said logic products during a second
- 7 testing period following said first testing period, said second testing technique
- 8 comprising loading said input latches by repeating test vectors of a minimum set
- 9 of test vectors obtained by repeating a last care bit in neighboring non-care bit
- 10 positions.

11

- 1 22. The method of claim 21, wherein said first testing period is approximately
- 2 10% of the test patterns testing approximately 95% of the faults, and said second
- 3 testing period is approximately 90% of the test patterns testing approximately 5%
- 4 of the faults.
- 1 23. The method of claim 21, wherein said first testing period consists of tests
- 2 sufficient for achieving a certain fault coverage threshold that is lower than the
- 3 total fault coverage objective for the full testing period, and said second testing
- 4 period consists of additional tests sufficient to raise the fault coverage to the total
- 5 fault coverage objective for the full testing period.
- 1 24. The method of claim 23 wherein said second test period would contain
- 2 several more tests than said first period.